

The Nature of the Isosurface Fractal Dimension

Marc Khoury

June 2012

Department Computer Science & Engineering
The Ohio State University
Columbus, OH 43210

Thesis Committee:

Tamal Dey, Professor
Yusu Wang, Associate Professor
Rephael Wenger, Associate Professor, Advisor

*Submitted in partial fulfillment of the requirements
for Graduation with Honors Research Distinction
in Computer Science & Engineering at The Ohio State University.*

Keywords: Isosurfaces, scalar data, fractal dimension, local fractal dimension, filters.

Dedicated to my parents, Antoinette and Isaac Khoury.

Abstract

A 3D scalar grid is a grid of vertices where each vertex is associated with some scalar value. The grid covers some rectilinear region and partitions that region into cubes. An isosurface for a given isovalue is a triangular mesh which approximates the level set for the isovalue. The size of an isosurface is the number of triangles in the isosurfaces mesh, while the size of a scalar grid is the number of cubes. The relationship between these two quantities describes the complexity of the isosurface. We introduce the fractal dimension of an isosurface and show that it is a powerful metric for describing the complexity of an isosurface.

Computing the isosurface fractal dimension for 60 benchmark data sets, we determine the average growth rate of an isosurface mesh. The number of connected components in an isosurface mesh gives a measure of the topological noise in the data set. We show that there is a high correlation between the fractal dimension and topological noise present in isosurface meshes. To better describe the relationship between noise and fractal dimension, we employ probabilistic methods to derive a formula for the fractal dimension as a function of uniform noise present in a data set.

We can restrict our definition of isosurface fractal dimension to a small local region and compute the fractal dimension for a local region around each vertex. The local isosurface fractal dimension gives a measure of the complexity of the isosurface in small regions of the grid. This method can be used for identification of noisy regions in an isosurface mesh. Noise filtering techniques can take advantage of this identification technique to more effectively remove noise from scalar data. Lastly we present an isosurface construction algorithm that moves the isosurface away from noisy regions in the grid to produce a smooth mesh.

Acknowledgments

First and foremost I would like to thank my advisor, Dr. Rephael Wenger. Surely this thesis would not have been possible without your brilliance and guidance. Thank you for showing me the beauty behind geometric algorithms and sparking my interest in research. Thank you for your patience as I worked through countless pressing trials and for your encouragement to never give up. Your mentorship these past few years has been invaluable and I truly enjoyed every minute of it.

I would like to thank the members of my thesis committee, Drs. Tamal Dey and Yusu Wang. Thank you for taking the time to review this thesis. I learned a lot about algorithms, computability, topology and geometry from the both of you. The classes that I took with the both of you were some of the most enjoyable of my undergraduate career. You are two of the most impressive researchers and mentors that I will ever have the privilege of learning from.

I would like to thank Dr. Kenneth Supowit for allowing me to spend countless hours chatting in his office. I enjoyed every conversation.

To my mentors at AT&T Labs, thank you for one of the best summers of my life. In particular I would like to thank Dr. Carlos Scheidegger for being an amazing mentor and friend. Carlos, you are one of the smartest and most knowledgeable people I know and I learned a great deal from you in the short time we have worked together. I hold your advice on both academic and non-academic matters in the highest regard and I appreciate that I can always turn to you when I need guidance. Thanks to Dr. Hamish Carr for his insight on local fractal dimension. I would also like to thank Dr. Craig Jackson for teaching me a great deal about abstract algebra and topology. You have made every other branch of mathematics I have encountered seem easy relative to quantum braid group representations.

Over the past few years I have made some amazing friends that have helped me grow as a person and whose continuous support I greatly appreciate. Thanks to James Austrow, Larry Capuder, David Goulder, Dorian Rahamim and Kelly Yarber for making sure I always had an outlet to relieve stress and that I had fun while in college. Thanks to Michael Schoenberg for teaching me a great deal about other areas of computer science and for your excellent advice throughout the years. Angela Dedy deserves a special thank you for keeping my ego in check for the first few years of undergrad. Thanks to Dominic Labanowski whose healthy rivalry continuously pushed me to work harder. Last, but certainly not least, I would like to thank Katie Kuksenok for helping me understand the importance of using one's gifts to benefit the world. My conversations with you greatly expanded my world view and shaped many of the opinions I hold today.

I would like to thank my parents, Antoinette and Isaac Khoury, as well as my brother, Jason Khoury, for their support. Thank you for encouraging me to pursue my dreams and doing everything you could to help me along the way. No matter what situation I find myself in, I know I can always count on you to help me through it. I dedicate this thesis to you.

Contents

1	Introduction	1
2	Fractal Dimension	5
2.1	Hausdorff Dimension	5
2.1.1	Measures	5
2.1.2	Hausdorff Measure	7
2.1.3	Hausdorff Dimension	7
2.2	Box-Counting Dimension	7
2.3	Isosurface Fractal Dimension	8
2.4	Local Isosurface Fractal Dimension	11
3	Noise and Fractal Dimension	15
3.1	Correlating Noise and Fractal Dimension	15
3.2	Fractal Dimension as a Function of Noise	17
3.2.1	Experimental Results	23
4	Filtering	25
4.1	Median Filtering	25
4.2	Gaussian Filtering	26
4.3	Fractal Filtering	26
5	Smooth Isosurface Construction	29
5.1	Experimental Results	30
6	Conclusion	33
6.1	Intrinsic Fractal Dimension	33
6.2	Fractal Formula	33
6.3	Filtering	34
6.4	4D Fractal Dimension	34
	Bibliography	35

List of Figures

2.1	Tooth data set (www.stereofx.org): Fractal box span dimension as a function of isovalue. (Tooth data set provided by GE Aircraft Engines, USA.)	9
2.2	Histogram of average fractal dimensions for 60 benchmark data sets.	10
2.3	Isosurface for the tooth data set for isovalue 850. The surface on the right is colored according to local fractal dimension. High fractal regions are colored in red and low fractal regions are colored in blue.	13
2.4	An isosurface for an aneurism data set. The isosurface is very noisy so much of its surface is colored red including most of the tiny disconnected components. . .	13
3.1	Noisy isosurfaces from the Tooth data set.	15
3.2	Tooth data set: Number of isosurface components as a function of isovalue (log scale).	16
3.3	Histogram of correlation between average fractal dimension and number of isosurface components for 65 benchmark data sets.	16
3.4	Plot of fractal formula (Equations 3.2, 3.3 and 3.4.) Function $f(r)$ represents the fractal edge span dimension as a function of uniform noise in the range $[-r\gamma, r\gamma]$ where γ is the gradient. Plot of fractal edge span dimension and fractal box span dimension versus noise added to a point cloud data set.	18
3.5	Plot of fractal edge span dimension versus noise added to a data set representing Euclidean distances to a set of 10 random points. Plot of fractal edge span dimension versus noise added to a data set representing distance to a circle. Plot of fractal formula.	23
3.6	Plot of fractal dimension (isovalue 650) vs. noise added to the Tooth data set and plot of fractal formula adjusted for the gradient 120.	24
3.7	Plot of fractal dimension (isovalue 65) vs. noise added to the Boston Teapot data set and plot of fractal formula adjusted for the gradient 25.	24
4.1	An isosurface for isovalue 850 extracted from the Tooth data set before(left) and after(right) a median filter was applied.	25
4.2	An isosurface for isovalue 20 extracted from the Aneurism data set before(left) and after(right) a median filter was applied. Since the entire grid was filtered, a great amount of detail was lost in the capillaries.	26

4.3	Using the local fractal dimension to identify the noisy regions in a scalar grid, scalar grid vertices are marked for filtering. A median filter is then applied only to the marked vertices. While only a portion of the grid was filtered there is no apparent decrease in quality in the isosurface.	27
4.4	The local fractal dimension was used to identify the noise in the scalar grid and a median filter was applied only in the noisy regions. The resulting isosurface is much smoother than the original but preserves much more detail in the capillaries than if we filtered the entire grid.	27
5.1	Algorithm 2 applied to the Tooth data set at isovalue 850. Various values for the parameter h determine how much of the surface is removed. From left to right these values are 2.1, 2.2, and 2.5.	31
5.2	Algorithm 2 applied to the VisMale data set at isovalue 70. The image on the left is the isosurface generated by the Marching Cubes algorithm for VisMale at isovalue 70. The center image applies Algorithm 2 with h parameter 2.2. The right image uses value 2.5.	31

List of Tables

2.1	Measurements of fractal box dimension and correlation between fractal dimension and number of isosurface components.	11
-----	--	----

Chapter 1

Introduction

An *isosurface* is an approximation to a level set of some scalar function in \mathbb{R}^3 . Applications, such as MRI and CT scanners, produce regular $n_1 \times n_2 \times n_3$ *scalar grids* where each vertex is associated with some scalar value s_v . Applying trilinear interpolation to each grid cube gives a scalar function on each cube. Combining these functions across all cubes gives a scalar function $f : \Omega \rightarrow \mathbb{R}$. An *isosurface* with *isovalue* σ is a triangular mesh which approximates the level set $f^{-1}(\sigma)$.

The Marching Cubes algorithm [6] extracts a isosurface mesh from a scalar grid. The algorithm marches through the scalar grid examining each grid cube once. The eight grid vertices are categorized based on the isovalue. Based on the configuration a set of triangles is selected from a lookup table. Combining the triangles generated for all the grid cubes creates an isosurface mesh that approximates the level set of the scalar function for the isovalue.

The size of an isosurface mesh can be measured in terms of its area or the number of triangles. The problem with these metrics is that they depend upon the specific approximation to $f^{-1}(\sigma)$. A better measurement is to count the number of grid cubes intersected by the isosurface. More specifically count the number of grid cubes with span $[\alpha, \beta]$ such that $\alpha < \sigma \leq \beta$. For isosurfaces constructed by the Marching Cubes algorithm, and numerous variants, the number of grid cubes where $\alpha < \sigma \leq \beta$ is exactly the number of grid cubes intersected by the isosurface. For such isosurfaces, isosurface area and the number of isosurface triangles are directly proportional to the number of intersected grid cubes.

Increases in the resolution of scanning devices permit increases in the size of the scalar grid. As the size of the grid grows so does the size of the isosurface with isovalue σ . The relationship between the size of the isosurface and the size of the scalar grid depends upon how “space-filling” the isosurface is. To measure the complexity of an isosurface we employ the *fractal dimension*, which is a measure of how much a point set scales with respect to the space around it.

Let X be a subset of a rectilinear region Ω . Let $I_N(X)$ be the number of cubes intersected by X when Ω is covered by a grid of N cubes with dimensions $\delta \times \delta \times \delta$. The *fractal box counting dimension* of X is

$$\lim_{\delta \rightarrow 0} \frac{\log(I_N(X))}{\log(1/\delta)}.$$

The volume of Ω is $N\delta^3$. Since the volume is constant, δ is proportional to $1/N^{1/3}$. The fractal

box counting dimension can be redefined as

$$\lim_{\delta \rightarrow 0} \frac{\log(I_N(X))}{\log(1/\delta)} = \lim_{N \rightarrow \infty} \frac{\log(I_N(X))}{\log(N^{1/3})} = \lim_{N \rightarrow \infty} 3 \frac{\log(I_N(X))}{\log(N)}. \quad (1.1)$$

The fractal box counting dimension represents the growth in the number of intersected by X as a function of the grid size. While isosurfaces are not simply subsets X of Ω they are similar enough that we can define an equivalent definition of fractal box counting dimension. Instead of taking the limit as N goes to infinity, we examine the scalar grid at full resolution and at a subsampled resolution. Comparing the number of intersected cubes for both grids gives a measure of the fractal dimension.

The fractal dimension of an isosurface is directly related to the growth rate of the isosurface. A fractal dimension of 2 corresponds to a growth rate of $\Theta(N^{2/3})$. Similarly, a fractal dimension of 3 corresponds to a growth rate of $\Theta(N)$. More generally, a fractal dimension of 3κ with $2/3 \leq \kappa \leq 1$ corresponds to a growth rate of $\Theta(N^\kappa)$.

The fractal dimension for an isosurface from a scalar grid that was generated by sampling an algebraic function $f : \mathbb{R}^3 \rightarrow \mathbb{R}$ is two. Similarly the fractal dimension of any nearly planar surface is two. However what is the average fractal dimension of isosurfaces produced by benchmark data sets?

In previous work, [3], it had been suggested that the number of intersected cubes grows at a rate of $\Theta(N^{0.82})$. A subsequent paper, [10], reported a growth rate of $\Theta(N^{0.96})$. We compute the average fractal dimension of 60 benchmark data sets, the same data sets used in [3] and [10], and conclude that the average growth rate is substantially less than previously suggested.

The fractal dimension for a scalar grid can be plotted as a function of the isovalue using the techniques from [1]. The resulting fractal dimension plot proves to be very useful for isovalue selection and understanding any existing relationships with other measurable quantities. Isovalues with fractal dimension near two tend to produce smooth isosurfaces while those with high fractal dimension produce noisy isosurfaces with many small connected components.

It was also shown in [3] and [10] that the growth rate increases with the noise in the scalar grid. We wish to demonstrate a correlation between the noise in the scalar grid and the fractal dimension for benchmark data sets. *Topological noise* refers to small topological features, including loops and tunnels, that are caused by noise in the scalar grid. To measure the topological noise we plot the number of connected components as a function of the isovalue for each of our 60 benchmark data sets. We conclude that there is a strong positive correlation between the number of connected components and the fractal dimension.

The fractal dimension increases with noise. We derive an analytic formula which predicts the fractal dimension as a function of the noise in the grid. Assuming a uniform noise model, our formula is based on the probability that an edge is intersected when a varying amount of noise is added to the vertices. We confirm our formula with experimental results using both synthetic and benchmark data sets.

There exist fractal that exhibit different behavior in different regions of their point sets. The dimension of one part of the fractal may differ from some other part. Turning our attention more locally, we define the *local isosurface fractal dimension*. The local isosurface fractal dimension computes the complexity of a small region around a grid vertex. We demonstrate that the local

isosurface fractal dimension can be used for noise identification and then applied to filtering techniques.

The most common filtering techniques apply a smoothing operation to each grid vertex. For example a median filter will replace each vertex with the median of its neighbors and a Gaussian filter will compute a weighted average based on the neighboring values. Such filtering techniques are very effective on scalar grid data but do not utilize any information about the data set. By computing the local isosurface fractal dimension and identifying noisy regions in the scalar grid, we selectively filter the vertices with high local fractal dimension.

Another application of this noise identification technique is as an aid in isosurface construction. For a particular isovalue, we identify the noisy regions in the scalar grid and locate the intersected grid edges with high local fractal dimension. Adjusting the scalar values of these intersected edges moves the isosurface through the grid. By adjusting these scalar values we move the isosurface to a less noisy region. An isosurface constructed from the adjusted grid is smooth and contains less topological noise.

To summarize, the contributions of this thesis are as follows.

1. Definition of the fractal box span dimension which can be easily computed and plotted as a function of scalar values.
2. Definition of the local fractal box span dimension which can be used to identify noisy regions in a scalar grid for a particular isovalue.
3. Measurements of the isosurface growth and fractal dimension of benchmark data sets using grid subsampling.
4. Correlation of the fractal dimension with topological noise.
5. An analytic formula for the fractal dimension as a function of uniform noise.
6. Application of local isosurface fractal dimension to noise filters.
7. A isosurface construction algorithm which utilizes the local fractal dimension to produce smooth isosurfaces.

Chapter 2

Fractal Dimension

Fractals are mathematical objects that exhibit several very unique properties. The first is *self-similarity* which means as we zoom into a small region of the fractal we tend to find ourselves right back where we started even though we are only looking at a small subset of the original fractal. There is an infinite level of detail. Another unique property is that fractals are *space filling*. They occupy more space than their topological dimension implies. A fractal in a two dimensional space may occupy the entire two dimensional space or just some fraction of it that is greater than one.

Fractals usually have dimension that exceed their topological dimension, referred to as *fractal dimension*. The idea behind fractal dimension was introduced by [7] in an attempt to measure coastal lines. It was observed that as the size of the measuring stick used to measure the decreased the length of the coast increased. The coast exhibited self-similar structure and its complexity could be measured by a quantity that had many properties of a dimension although it was fractional.

Fractal dimension is unique from standard notions of dimension because it need not be integer. As we will see there are many ways to define fractal dimension and not all are equivalent.

2.1 Hausdorff Dimension

The Hausdorff dimensions generalizes the notion of dimension to non-integer values. The Hausdorff dimension is defined for any set and is based on measures which are easy to work with mathematically. Familiar geometric objects, such as lines and planes, have Hausdorff dimension equal to their topological dimension. However many interesting sets have non-integer Hausdorff dimension.

2.1.1 Measures

Measures seek to place a numeric value on the size of a set. In our work we only need to worry about measures on sets that are subsets of \mathbb{R}^n .

Definition 1. A function μ is a measure on \mathbb{R}^n if μ maps every subset of \mathbb{R}^n to a non-negative value, possibly ∞ , such that:

1. $\mu(\emptyset) = 0$
2. $\mu(A) \leq \mu(B)$ if $A \subset B$
3. if A_1, A_2, \dots is a countable sequence of sets then

$$\mu\left(\bigcup_{i=0}^{\infty} A_i\right) \leq \sum_{i=0}^{\infty} \mu(A_i)$$

with equality when A_i are disjoint Borel sets.

We think of the value $\mu(A)$ as the size of the set A and refer to $\mu(A)$ as the *measure* of A . Condition 1 states that the empty set must have measure zero, as it is the smallest possible set. Condition 2 says that if A is a subset of B then the measure of B must be at least as large as the measure of A . Finally the last condition states that for any countable sequence of, possibly overlapping, sets, the measure of their union must be less than or equal to the sum of their measures. Equality in the last condition occurs when the sequence A_i are disjoint Borel sets.

Examples

To aid in the understanding of measure, we present a simple example followed by a very important measure.

Example 1. Let A be a subset of \mathbb{R}^n and let $\mu(A)$ be the number of elements in A . The function μ is known as the counting measure.

The number of elements in a set create a measure on all subsets of \mathbb{R}^n . It is very easy to show that all the properties of a measure hold for the counting measure.

Lebesgue Measure

The Lebesgue measure \mathcal{L}^1 defines the length of a wide variety of subsets of \mathbb{R} . Consider any open subset (a, b) of \mathbb{R} and the corresponding closed set $[a, b]$ for some $a \leq b$. The Lebesgue measure $\mathcal{L}^1((a, b)) = \mathcal{L}^1([a, b]) = b - a$. Generalizing this notion to a set $A = \bigcup_i [a_i, b_i]$ where the intervals $[a_i, b_i]$ are some countable collection, we have $\mathcal{L}^1(A) = \sum_i (b_i - a_i)$. Thus the length of A is the sum of the length of its intervals. Finally we define the Lebesgue measure \mathcal{L}^1 as

$$\mathcal{L}^1(A) = \inf \left\{ \sum_{i=0}^{\infty} (b_i - a_i) : A \subset \bigcup_{i=0}^{\infty} [a_i, b_i] \right\}.$$

Let $A = \{(x_1, x_2, \dots, x_n) \in \mathbb{R}^n : a_i \leq x_i \leq b_i\}$ be a parallelepiped in \mathbb{R}^n . The n -dimensional volume of this set is $\text{vol}^n(A) = \prod_{i=1}^n (b_i - a_i)$. The Lebesgue measure on \mathbb{R}^n , denoted \mathcal{L}^n is defined as

$$\mathcal{L}^1(A) = \inf \left\{ \text{vol}^n(A_i) : A \subset \bigcup_{i=0}^{\infty} A_i \right\}.$$

The Lebesgue measure defines a measure on subsets of \mathbb{R}^n and can be thought of as a n -dimensional volume.

2.1.2 Hausdorff Measure

Given any subset of n -dimensional Euclidean space, the *diameter* of the set is the furthest distance between any two points of the set. Formally we define the diameter of a set A as

$$|A| = \sup\{|x - y| : x, y \in A\}.$$

A δ -cover of a set X is collection of sets $\{U_i\}$ such that $X \subset \bigcup_i U_i$ and we have $0 \leq |U_i| \leq \delta$. The *Hausdorff measure* of a subset X of \mathbb{R}^n for some non-negative number s and $\delta > 0$ is

$$\mathcal{H}_\delta^s(X) = \inf\left\{\sum_i |U_i|^s : \{U_i\} \text{ is a } \delta\text{-cover of } X\right\}.$$

The larger the value of δ , the more possible covers of X exist. Taking the limit as $\delta \rightarrow 0$, the value of \mathcal{H}_δ^s approaches some value, possibly ∞ . The s -dimensional Hausdorff measure is defined as

$$\mathcal{H}^s(X) = \lim_{\delta \rightarrow 0} \mathcal{H}_\delta^s(X).$$

2.1.3 Hausdorff Dimension

Now that we have formally defined the Hausdorff measure it is very easy to introduce the Hausdorff dimension. The *Hausdorff dimension* of a set $X \subset \mathbb{R}^n$ is defined as

$$\dim_H(X) = \inf\{s \geq 0 : \mathcal{H}^s(X) = 0\} = \dim_X = \sup\{s : \mathcal{H}^s(X) = \infty\}.$$

Taking the supremum of the empty set to be 0, the Hausdorff dimension is defined for any subset of \mathbb{R}^n .

Intuitively, this definition states that there is a particular value of s such that the Hausdorff measure jumps from 0 to ∞ . The value s is the dimension of the set X .

2.2 Box-Counting Dimension

The box-counting dimension, also known as the Minkowski-Bouligand dimension, is another way to define the fractal dimension of a set X in \mathbb{R}^n or, more generally, in a metric space. It is also the definition that most closely resembles our formulation of isosurface fractal dimension. To compute the box-counting dimension for a fractal X , consider X lying on a square grid and count the number of boxes that intersect X . The box-counting dimension measures how the number of intersected boxes changes as the size of the boxes decreases.

More formally, we define the fractal box-counting dimension for some subset of a rectilinear region in \mathbb{R}^3 . Let X be a subset of a rectilinear region Ω . Let $I_N(X)$ be the number of grid cubes intersected by X when Ω is covered by a grid of N cubes with dimensions $\delta \times \delta \times \delta$. The *fractal box-counting dimension* of X is [5]

$$\dim_B = \lim_{\delta \rightarrow 0} \frac{\log(I_N(X))}{\log(1/\delta)}. \quad (2.1)$$

The volume of Ω equals $N\delta^3$. Since the volume is constant, δ is proportional to $1/N^{1/3}$. The fractal box-counting dimension can be redefined as

$$\lim_{\delta \rightarrow 0} \frac{\log(I_N(X))}{\log(1/\delta)} = \lim_{N \rightarrow \infty} \frac{\log(I_N(X))}{\log(N^{1/3})} = \lim_{N \rightarrow \infty} 3 \frac{\log(I_N(X))}{\log(N)}. \quad (2.2)$$

The limit may not always be defined for a set X . If the limit does not exist one must compute the upper and lower box-counting dimensions which correspond to the upper and lower limit respectively. We denote these quantities with \dim_{UB} and \dim_{LB} respectively. The fractal box-counting dimension is well-defined only if the upper and lower box-counting dimensions are equal.

While not necessarily equal to the earlier defined Hausdorff dimension, the box-counting dimension and the Hausdorff dimension are related. The three are related by the following inequality

$$\dim_H \leq \dim_{LB} \leq \dim_{UB}. \quad (2.3)$$

For many fractals the three are equal. However the existence of fractals where this inequality is strict shows that the two definitions are not equivalent. Regardless, such a lovely relationship between the three is both surprising and useful. This inequality shows that the box-counting dimension must always be at least as high as the Hausdorff dimension.

2.3 Isosurface Fractal Dimension

While isosurfaces are not simply subsets X of some rectilinear region Ω , they are similar enough to define an equivalent box-counting definition. Instead of taking the limit as N goes to infinity, we subsample the scalar grid and compare the number of intersected cubes in the full resolution grid to the subsampled grid.

There are many algorithms for isosurface construction which produce numerous, albeit similar, isosurfaces. The sizes of these isosurfaces differ only by constant factors so all have the same asymptotic growth rate. We consider isosurface generated by the Marching Cubes algorithm [6].

The *span* of a scalar grid cube C is a closed interval $[\alpha, \beta]$ where α is the smallest scalar value of any cube vertex and β is the largest. Any isosurface with isovalue σ will intersect C if σ lies strictly between α and β . If σ is equal to α or β then the isosurface may or may not intersect C . For a scalar grid G , let $I^\sigma(G)$ be the number of cubes whose span contains σ .

Let $f : \Omega \rightarrow \mathbb{R}$ be a scalar function defined on a rectangular region Ω . Function f is not necessarily continuous. Let G_N be a scalar grid with N cubes covering Ω which samples f at its vertices. Define the *fractal box span dimension* of f for isovalue σ as

$$\lim_{N \rightarrow \infty} \frac{\log(I^\sigma(G_N))}{\log(N^{1/3})} = \lim_{N \rightarrow \infty} 3 \frac{\log(I^\sigma(G_N))}{\log(N)}. \quad (2.4)$$

If this limit does not exist, then we should use the upper and lower limits.

The fractal box span dimension is similar to the fractal box counting dimension defined in section 2.2. However it is not necessarily the same as the fractal box-counting dimension of the

level set $f^{-1}(\sigma)$. If f is not continuous, there may be no point p for which $f(p)$ equals σ and yet an infinite number of arbitrarily close pairs of points (p, q) such that $f(p) \leq \sigma \leq f(q)$.

There are multiple methods that have been proposed to measure the fractal box-counting dimension of a point set X . The most common of these techniques involves plotting $3 \log(I_N(X))$ as a function of $\log(N)$ and computing the slope of the linear regression line fitting the points. The slope is the dimension of the set. Alternatively, Pekar et. al. [9] measure $3 \log(A^\sigma(G_N))$ as a function of $\log(N)$ and take the slope of that linear regression line, where A^σ is the area of the isosurface with isovalue σ .

Instead we consider only two grids, G_N and $G_{N/8}$, where $G_{N/8}$ has $N/8$ cubes. The slope through the points $(\log(N), \log(I^\sigma(G_N)))$ and $(\log(N/8), \log(I^\sigma(G_{N/8})))$ is

$$\frac{\log(I^\sigma(G_N)) - \log(I^\sigma(G_{N/8}))}{\log(N) - \log(N/8)} = \frac{\log(I^\sigma(G_N)) - \log(I^\sigma(G_{N/8}))}{\log(2)} = \log_2\left(\frac{I^\sigma(G_N)}{I^\sigma(G_{N/8})}\right).$$

Let G be a scalar grid with N cubes and let G' be the subsampled grid of G with $N/8$ cubes. The *fractal box span dimension* for a scalar grid G with isovalue σ is defined as

$$\log_2\left(\frac{I^\sigma(G)}{I^\sigma(G')}\right) \quad (2.5)$$

There are many advantages to our definition. Unlike previous definitions it uses only two grids and does not require an algorithm to compute a linear regression line. It is a simple and precise formula. Our definition uses the number of intersected cubes for a particular isovalue instead of the isosurface area. It does not depend on any isosurface construction algorithm, does not require constructing an isosurface, and can be computed easily by only examining the scalar grid.

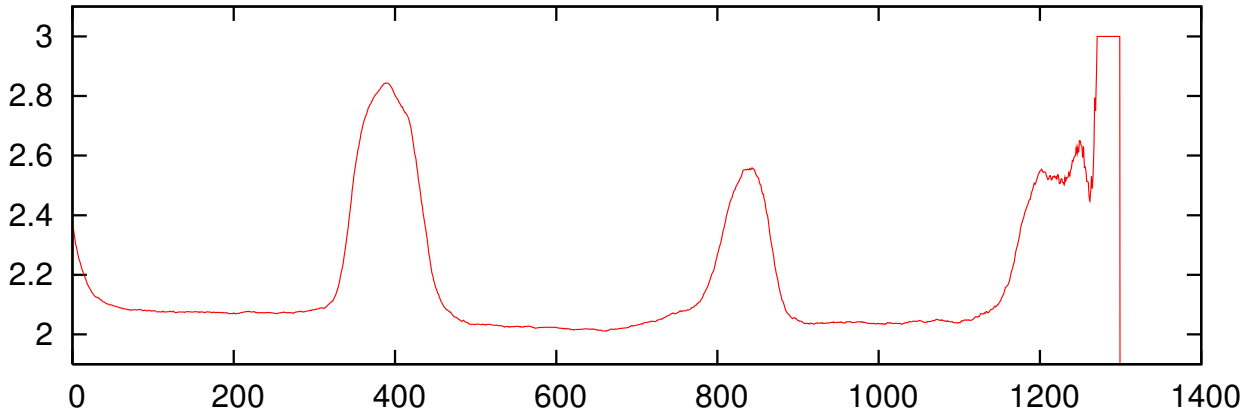


Figure 2.1: Tooth data set (www.stereofx.org): Fractal box span dimension as a function of isovalue. (Tooth data set provided by GE Aircraft Engines, USA.)

We measured the fractal box span dimension of 60 public benchmark data sets available from www.stereofx.org and www.volvis.org. Each data set represents a scalar grid,

where the scalar values are 8 or 12 bit integers. For each scalar grid we computed the fractal box span dimension for each scalar value, from the minimum to the maximum (see Figure 2.1), and computed an average fractal dimension for each grid. The average fractal dimension for all the data sets is 2.26 with a standard deviation of 0.16 (see Figure 2.2). We also computed the standard deviation of the fractal dimension for each data set. The average standard deviation is 0.13.

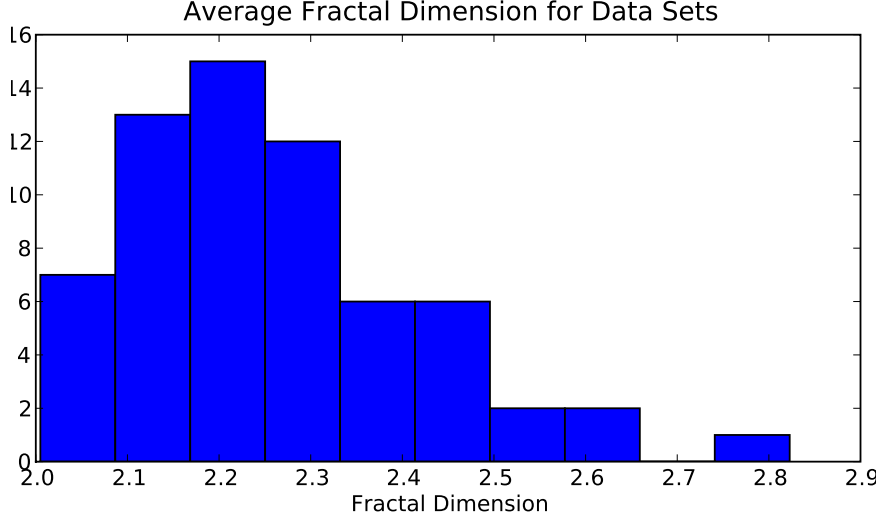


Figure 2.2: Histogram of average fractal dimensions for 60 benchmark data sets.

The growth rate of an isosurface is directly related to fractal dimension. If the number of intersected cubes grows as $\Theta(N^{2/3})$ then the fractal dimension is 2. If the number of intersected cubes grows as $\Theta(N)$ then the fractal dimension is 3. More generally if the number of intersected cubes grows as $\Theta(N^\kappa)$ for some $0 \leq \kappa \leq 1$ then the fractal dimension is 3κ .

An average fractal dimension of 2.26 gives an isosurface growth rate of $\Theta(N^{0.75})$. This is much less than the previously proposed growth rates of $\Theta(N^{0.82})$ [3] or $\Theta(N^{0.96})$ [10]. As was done in [3] and [10] we partitioned the data sets into three distinct types, medical, measured but non-medical, and synthetic, and measured the average fractal dimension of each type. In [10] the measured growth rates for each type were reported as $\Theta(N^{0.70})$, $\Theta(N^{0.82})$, $\Theta(N^{0.87})$ for medical, measured, and synthetic receptively. Our corresponding measurements are $\Theta(N^{0.77})$, $\Theta(N^{0.75})$, $\Theta(N^{0.73})$.

Our measurements for the isosurface growth rate are much smaller than previously reported values. Additionally we witnessed little variation between different types of scalar grids in our measurements. This was not the case in [3] and [10].

The fractal box span dimension of a scalar grid is meant to approximate an infinite sequence of scalar grids. To determine how well it accomplishes this, we subsampled 60 scalar grids and measured the fractal dimension of these subsampled grids. The average fractal dimension of the subsampled grids is 2.38 with a standard deviation of 0.16. The average magnitude of the difference of the fractal dimension between the full resolution and subsampled grid is 0.15 with a standard deviation of 0.18.

Table 2.1: Measurements of fractal box dimension and correlation between fractal dimension and number of isosurface components. a) Type of data sets. Data sets of type “medical” are MRI and CT scans of humans. Data sets of type “measured” include non-organic data sets such as engine and organic data sets such as bonsai, monkey-CT and lobster. All other data sets are of type “synthetic”, and include numerical simulation data such as bluntfin, neghip or shockwave and computer generated models such as hydrogenAtom, nucleon or silicium. b) Number of data sets. c) Average fractal dimension. (Average of the average fractal dimension of each data set.) d) Standard deviation of fractal dimension. e) Average of the standard deviation of the fractal dimension. (Average of the standard deviation of the fractal dimension of each data set.) f) Average fractal dimension of subsampled data sets. g) Average of the magnitude of the difference between the fractal dimension of the full resolution and subsampled version of each data set. h) Average correlation between the fractal dimension and the number of connected components for each data set. i) Standard deviation of the correlation.

Data Sets		Fractal Box Dim					Correlation	
a) Type	b) Num	c) Avg	d) Std Dev	e) Avg Std Dev	f) Avg Subsampled	g) Avg Full – Sub	h) Avg	i) Std Dev
All	60	2.26	0.16	0.13	2.38	0.14	0.61	0.18
Medical	24	2.29	0.12	0.14	2.42	0.13	0.65	0.11
Measured (non-medical)	22	2.26	0.17	0.13	2.39	0.15	0.61	0.19
Synthetic	14	2.20	0.22	0.09	2.30	0.13	0.52	0.25

2.4 Local Isosurface Fractal Dimension

Many surfaces exhibit high fractal dimension in a few regions of the surface but low fractal dimension in others. Such objects are called *multifractals* and require a set of fractal dimensions to fully specify their behavior. To identify high fractal regions in scalar grids we focus on computing the isosurface fractal dimension locally around each vertex.

Let G be a scalar grid with N cubes and let G' be the subsampled grid of G with $N/8$ cubes. Let W be a window of size $k \times k \times k$ centered at a vertex v in the grid G . Let W' be a window of size $k/2 \times k/2 \times k/2$ that covers the same region as W but in the subsampled grid G' . Finally let $I_W^\sigma(G)$ denote the number of intersected cubes in G for isovalue σ restricted to the window W . The *local fractal box span dimension* for a vertex v is defined as

$$\log_2\left(\frac{I_W^\sigma(G)}{I_{W'}^\sigma(G')}\right) \quad (2.6)$$

The local fractal box span dimension is a restriction of our previous definition to a small region around a vertex in the grid. We compute the fractal dimension in this region as we would with a full grid. The local fractal dimension gives a measure of the complexity of the isosurface locally around a vertex in the grid.

For every grid vertex v the number of intersected cubes in an arbitrary $k \times k \times k$ region centered at v must be quickly computed. Additionally the number of intersected cubes in the subsampled region must also be computed. It is very easy to see that regions of neighboring vertices overlap heavily.

To quickly compute the number of intersected cubes we employ a summed area table [4]. A summed area table enables efficient queries for the sum of values in a rectangular subset of a

grid. The value at any point (x, y) in the summed area table is the sum of all the values above and to the left of (x, y) inclusive.

$$\text{sat}(x, y) = \sum_{\substack{i \leq x \\ j \leq y}} i(i, j)$$

The table can be computed in a single pass over each element in the input by using equation 2.7. This technique lends itself to a simple dynamic programming algorithm with time complexity $\Theta(N)$. Once the summed area table is constructed we can compute the sum of the values for *any* rectilinear region in constant time.

$$\text{sat}(x, y) = i(x, y) + \text{sat}(x - 1, y) + \text{sat}(x, y - 1) - \text{sat}(x - 1, y - 1) \quad (2.7)$$

It is easy to generalize summed area tables to three dimensions and apply them to isosurface grids by defining i as follows.

$$i(x, y, z) = \begin{cases} 1 & : \text{if the cube } (x, y, z) \text{ is intersected} \\ 0 & : \text{otherwise} \end{cases}$$

The recurrence for a three dimensional summed area table presented in equation 2.8 is a bit more complicated since it depends upon the values of eight cube vertices.

$$\begin{aligned} \text{sat}(x, y, z) = & i(x, y, z) + \text{sat}(x - 1, y, z) + \text{sat}(x, y - 1, z) + \text{sat}(x, y, z - 1) \\ & - \text{sat}(x - 1, y - 1, z) - \text{sat}(x, y - 1, z - 1) - \text{sat}(x - 1, y, z - 1) \\ & + \text{sat}(x - 1, y - 1, z - 1) \end{aligned} \quad (2.8)$$

Thus we can use a 3D summed area table to query the number of intersected cubes in *any* region in constant time. Since creation of the tables for both the original and subsampled grid takes $\Theta(N)$ time and computing a local fractal dimension takes a constant amount of time per vertex, the whole algorithm runs in $\Theta(N)$.

Once we compute a local fractal dimension for each vertex in a scalar grid, we can identify which regions in the grid are noisy for a given isovalue. To visualize the noisy regions we construct an isosurface mesh and assign a color to each isosurface vertex. The color is based on the local fractal dimension of the scalar grid vertices of the grid edge that the isosurface vertex lies on. The location of the vertex on the edge allows us to linearly interpolate a local fractal dimension for the vertex.

In our experiments we set every vertex with fractality below 2.2 to the color blue and every vertex above 2.5 to the color red. For fractal dimension between 2.2 and 2.5 a color is linearly interpolated between blue and red.

Figure 2.3 demonstrates the ability of local fractal dimension to detect noise in a scalar grid. The noisy portion of the tooth is entirely highlighted in red while the smooth portion is highlighted in blue. Figure 2.4 provides another example on an aneurism data set which contains many small components caused by noise in the scalar grid. Identifying the noise using local fractal dimension is the first step in our targeted filtering approach.

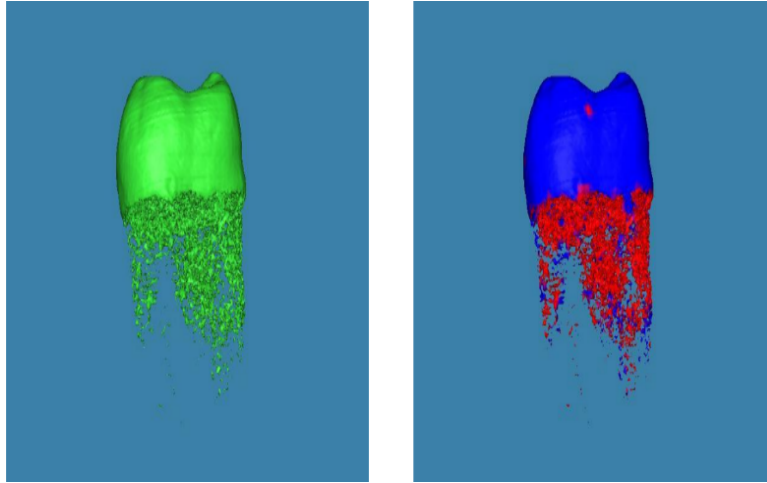


Figure 2.3: Isosurface for the tooth data set for isovalue 850. The surface on the right is colored according to local fractal dimension. High fractal regions are colored in red and low fractal regions are colored in blue.

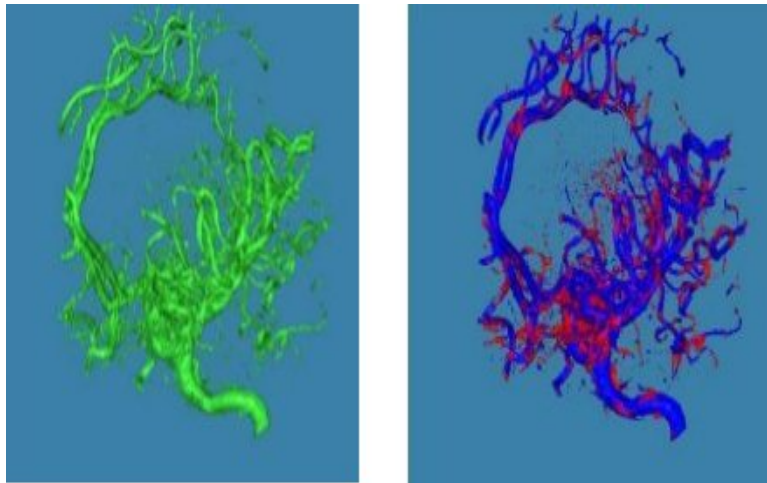


Figure 2.4: An isosurface for an aneurism data set. The isosurface is very noisy so much of its surface is colored red including most of the tiny disconnected components.

Chapter 3

Noise and Fractal Dimension

3.1 Correlating Noise and Fractal Dimension

The fractal box span dimension for a scalar grid varies greatly as a function of isovalue. Examining the isosurfaces with the largest fractal dimension it seems as though they are among the noisiest isosurfaces in a data set. Figure 3.1 shows two isosurfaces with isovalues 400 and 850 generated from the Tooth data sets. Both isosurfaces are extremely noisy and correspond to the peaks of the fractal dimension plot.

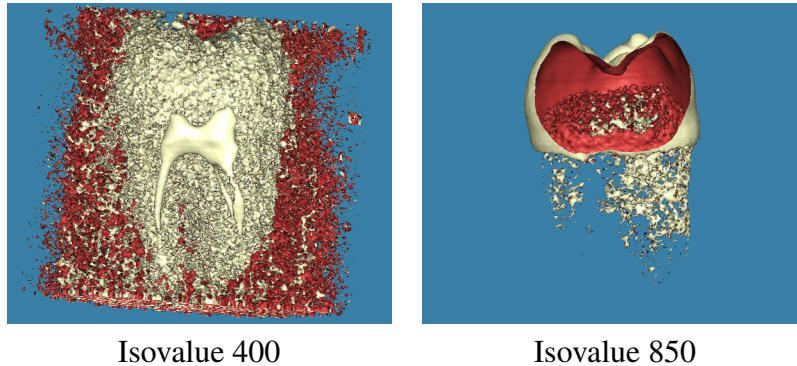


Figure 3.1: Noisy isosurfaces from the Tooth data set.

There has been work to show that the fractal dimension of an isosurface increases as the noise increase. Scheidegger et. al. [10] added computer generated noise to synthetic data sets to demonstrate that the complexity increased as more noise was added. However their experiment does not conclusively prove that high fractal dimension is caused by noise in a scalar grid.

To determine the relationship between noise and fractality, a metric is needed to measure the noise in a isosurface. The *topological noise* in an isosurface is topological components such as holes, loops, and small connected components that are caused by noise in the scalar grid. To measure the topological noise we count the number of connected components using countour trees.

To compute the number of connected components for every isovalue quickly we used the method proposed by Carr et. al. [2] for building contour trees. Each contour tree edge represents

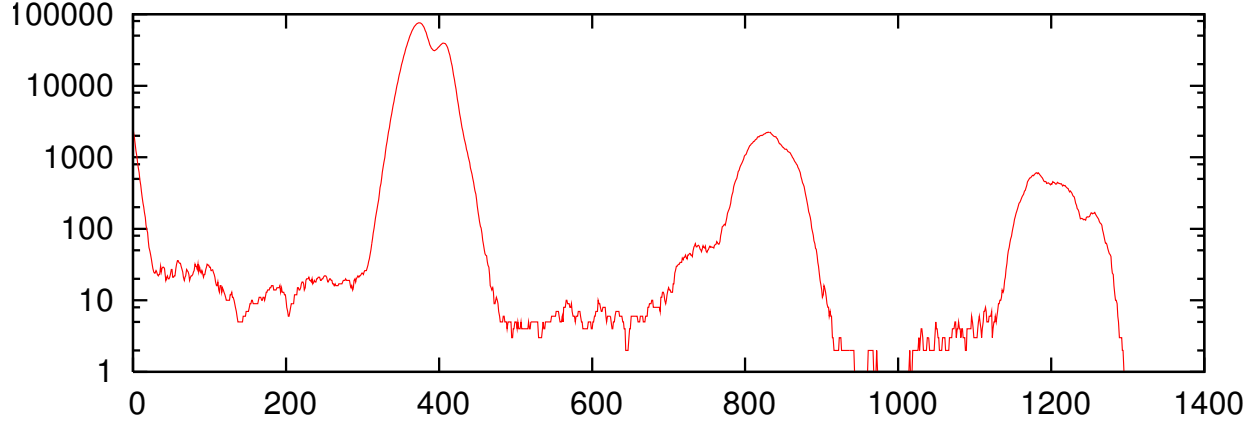


Figure 3.2: Tooth data set: Number of isosurface components as a function of isovalue (log scale).

a set of connected isosurface components which vary with the isovalue. For each isovalue we count the number of contour tree edges which represent an isosurface component with that isovalue. The total count equals the number of connected components for an isosurface mesh with that isovalue.

Both the fractal dimension and the number of connected components vary by isovalue. Examining Figure 3.2 we notice that the minimums and maximums of the graph approximately correspond to those of the fractal dimension plot in Figure 2.1. We computed the fractal dimension and connected components plots for 60 benchmark data sets. We then computed the correlation between the fractal dimension and connected components for each data set.

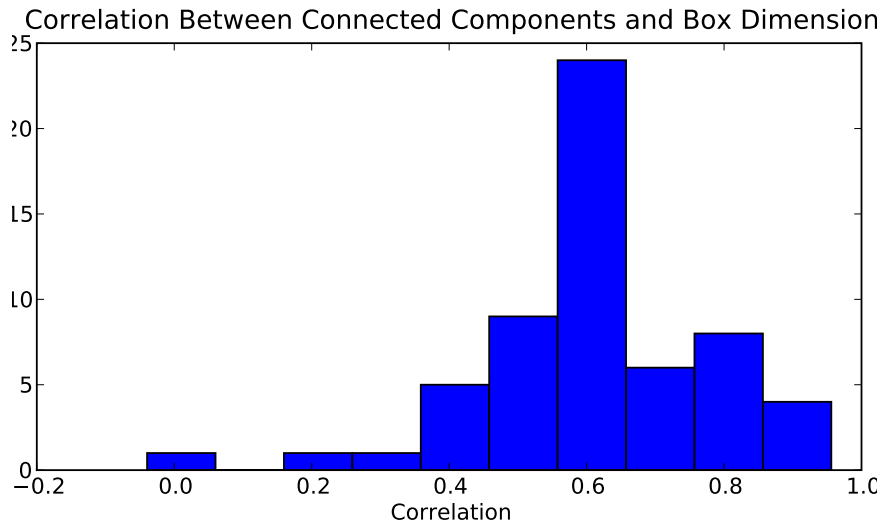


Figure 3.3: Histogram of correlation between average fractal dimension and number of isosurface components for 65 benchmark data sets.

The number of connected components in an isosurface is influenced by several factors, noise being just one. The isosurface area is another important factor that influences the number of connected components. Since the area of an isosurface differs across different isosurfaces we divide the number of connected components by the isosurface area to get a normalized measure of noise. The correlation between the resulting value and the fractal dimension was computed in our experiments. The isosurface area was approximated by the number of intersected edges. Isosurfaces with small area were ignored as they bias the result. To be more specific, we took the smallest two dimensional facets of the scalar grid and divided the product by four. Any isosurface that intersected less than that many edges was excluded from the measurement. Additionally we excluded four very simple synthetic data sets with fractal dimension very close to two regardless of isovalue and standard deviation less than 0.05 for the fractal dimension. Correlation statistics are presented in Table 2.1 and Figure 3.3.

3.2 Fractal Dimension as a Function of Noise

We wish to be able to predict the fractal dimension as a function of noise in the scalar data. Suppose noise from some noise distribution is added to some subset of the vertices of an otherwise smooth isosurface. What is the value of the fractal dimension after the noise has been added?

Uniform noise in a scalar grid is a set of random values selected from a uniform probability distribution that is added to a subset of the vertices in the grid. The particular probability distribution can have varying frequencies, but it is only high frequency noise that affects the fractal dimension.

Computing the probability that a grid cube is intersected by an isosurface after noise is added to an otherwise clean scalar grid is difficult. The analysis depends upon eight scalar values corresponding to the vertices of the cube. Instead we choose to examine a simpler model, the probability that an edge is intersected.

The *span* of a grid edge as $[\alpha, \beta]$ where α and β are the scalar values associated with the vertices of the edge and $\alpha \leq \beta$. Let $I_e^\sigma(G)$ denote the number of intersected edges in a grid G for isovalue σ . The *fractal edge span dimension* is defined as

$$\log_2\left(\frac{I_e^\sigma(G)}{I_e^\sigma(G')}\right) \quad (3.1)$$

The fractal edge span dimension differs from the previously defined fractal box span dimension as demonstrated by Figure 3.4. However the two definitions are similar.

Every grid vertex v has a corresponding scalar value s_v . Consider a perturbed \tilde{s}_v , which is s_v plus some uniformly distributed random value in the range $[-\mu, \mu]$. Now consider the grid \tilde{G} where each grid vertex s_v has been replaced by \tilde{s}_v . Our goal is to derive a formula that give the fractal dimension of \tilde{G} as a function of μ .

Clearly this is an oversimplified model, as most noise in real data sets does not come from uniform distribution. Gaussian noise would be a much more accurate representation of noise in real data sets. However predicting the fractal dimension as a function of noise is already a difficult problem and assuming a uniform distribution greatly simplifies the analysis.

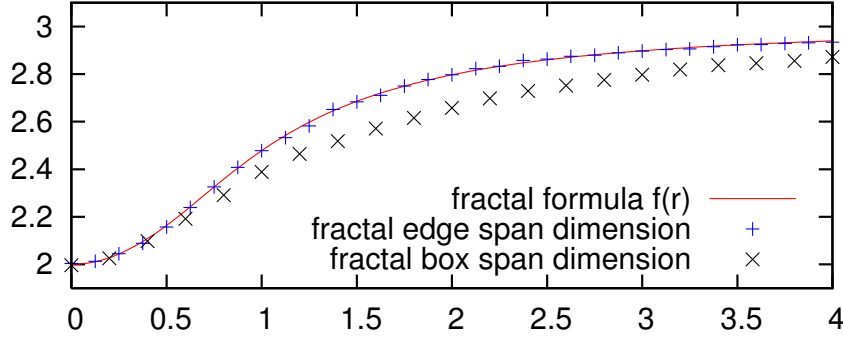


Figure 3.4: Plot of fractal formula (Equations 3.2, 3.3 and 3.4.) Function $f(r)$ represents the fractal edge span dimension as a function of uniform noise in the range $[-r\gamma, r\gamma]$ where γ is the gradient. Plot of fractal edge span dimension and fractal box span dimension versus noise added to a point cloud data set.

We place several conditions of the grid edges around an isosurface to aid in making any prediction on the fractal dimension of the isosurface. If edge e has edge span $[\alpha_e, \beta_e]$, (where $\alpha_e \leq \beta_e$, by convention,) then *the minimum scalar value* of edge e is α_e . The *magnitude of the span* of e is $|\beta_e - \alpha_e|$. Define a bivariate function Ψ on the grid edges as:

$$\Psi(e) = (\Psi_1(e), \Psi_2(e)) = (\alpha_e, |\beta_e - \alpha_e|).$$

Let σ be an isovalue and $E_\sigma(\eta)$ be the edges whose minimum scalar values are in the range $[\sigma - \eta, \sigma + \eta]$. Our formula will be based on the assumption that Ψ restricted to $E_\sigma(\eta)$ approximates a bivariate uniform distribution where $\Psi_1(e)$ varies uniformly in the range $[\sigma - \eta, \sigma + \eta]$ and $\Psi_2(e)$ varies uniformly from 0 to γ .

Let $E'_\sigma(\eta)$ be the edges of the subsampled grid G' whose minimum scalar values are in the range $[\sigma - \eta, \sigma + \eta]$. Under the appropriate conditions the relationship between the fractal dimension and the noise is given by the following theorem:

Theorem 1. *Let r and γ be non-negative real numbers and let η equal $(r + 2)\gamma$. Let G be a regular grid and σ be an isovalue such that Ψ restricted to $E_\sigma(\eta)$ and Ψ restricted to $E'_\sigma(\eta)$ approximates the bivariate uniform distribution on $[\sigma - \eta, \sigma + \eta] \times [0, \gamma]$ and where $|E'_\sigma(\eta)|$ approximately equals $|E_\sigma(\eta)|/8$. Let \tilde{G} be a scalar grid created by adding random noise with uniform distribution in the range $[-r\gamma, r\gamma]$ to G .*

1. *If $r \geq 1$, then the expected fractal edge span dimension of \tilde{G} is:*

$$\log_2 \left(\frac{32r^3 + 8r - 1}{32(r/2)^3 + 8(r/2) - 1} \right). \quad (3.2)$$

2. *If $1/2 \leq r \leq 1$, then the expected fractal edge span dimension of \tilde{G} is:*

$$\log_2 \left(\frac{32r^3 + 8r - 1}{r^4 + 6r^2} \right). \quad (3.3)$$

3. If $r \leq 1/2$, then the expected fractal edge span dimension of \tilde{G} is:

$$\log_2 \left(4 \frac{2r^2 + 3}{2(r/2)^2 + 3} \right). \quad (3.4)$$

All three parts of the formula are graphed in Figure 3.4. The value r is the ratio of the noise to the gradient. When r is zero, meaning that there is no noise in the grid, the expected fractal dimension is two. As the noise in the grid increases so does the expected fractal dimension.

A probabilistic argument is used to derive this formula. First the probability that an edge (v_a, v_b) is intersected in the grid \tilde{G} is computed (Lemma 1). The probability that an edge is intersected in \tilde{G} is dependent upon the original scalar values in G . Next, Lemma 2 compute the probability that an edge with gradient span δ is intersected by the isosurface. To do this we integrate over all minimum scalar values of edge with span magnitude δ . Finally in Lemma 3 we compute the probability that a random edge is intersected by the isosurface by integrating over the span magnitude δ . Based on these probabilities we can compute the expected number of intersected edges in both \tilde{G} and its subsampled grid \tilde{G}' . Applying the fractal edge span dimension definition gives the expected fractal dimension.

Lemma 1. *Let \tilde{G} be a scalar grid created by adding random noise with uniform distribution in the range $[-\mu, \mu]$ to G . Let (v_a, v_b) be a grid edge of G where v_a has scalar value s_a in G and \tilde{s}_a in \tilde{G} and v_b has scalar value s_b in G and \tilde{s}_b in \tilde{G} .*

1. *If $\sigma \in \mathbb{R}$ is a scalar value where $\sigma - \mu \leq s_a \leq s_b \leq \sigma + \mu$, then the probability that $\sigma \in [\tilde{s}_a, \tilde{s}_b]$ is: $\frac{1}{2} - \frac{(s_a - \sigma)(s_b - \sigma)}{2\mu^2}$.*
2. *If $\sigma \in \mathbb{R}$ is a scalar value where $s_a \leq \sigma - \mu \leq s_b \leq \sigma + \mu$, then the probability that $\sigma \in [\tilde{s}_a, \tilde{s}_b]$ is: $\frac{1}{2} + \frac{s_b - \sigma}{2\mu}$.*
3. *If $\sigma \in \mathbb{R}$ is a scalar value where $\sigma - \mu \leq s_a \leq \sigma + \mu \leq s_b$, then the probability that $\sigma \in [\tilde{s}_a, \tilde{s}_b]$ is: $\frac{1}{2} + \frac{s_a - \sigma}{2\mu}$.*

Proof. If $\sigma - \mu \leq s_a \leq \sigma + \mu$, then the probability that \tilde{s}_a is greater than σ is:

$$P(\tilde{s}_a \geq \sigma) = \frac{\mu + s_a - \sigma}{2\mu} = \frac{1}{2} + \frac{s_a - \sigma}{2\mu}.$$

Since s_a is in the range $[\sigma - \mu, \sigma + \mu]$, the value on the right is always between 0 and 1. The probability that s_a is less than σ is:

$$P(\tilde{s}_a \leq \sigma) = \frac{\mu - (s_a - \sigma)}{2\mu} = \frac{1}{2} - \frac{s_a - \sigma}{2\mu}.$$

Similar probabilities hold for s_b and \tilde{s}_b .

The probability that $\sigma \in [\tilde{s}_a, \tilde{s}_b]$ is the probability that $\tilde{s}_a \leq \sigma \leq \tilde{s}_b$ or that $\tilde{s}_a \geq \sigma \geq \tilde{s}_b$. Since these two events are mutually exclusive, (except for the zero probability event that $\tilde{s}_a = \tilde{s}_b = \sigma$.) the probability that $\sigma \in [\tilde{s}_a, \tilde{s}_b]$ is the sum of these two probabilities.

Case 1: $\sigma - \mu \leq s_a \leq s_b \leq \sigma + \mu$.

$$\begin{aligned} P(\sigma \in [\tilde{s}_a, \tilde{s}_b]) &= P(\tilde{s}_a \leq \sigma \leq \tilde{s}_b) + P(\tilde{s}_a \geq \sigma \geq \tilde{s}_b) \\ &= \frac{1}{2} - \frac{(s_a - \sigma)(s_b - \sigma)}{2\mu^2}. \end{aligned}$$

Case 2: $s_a \leq \sigma - \mu \leq s_b \leq \sigma + \mu$.

The probability that $\tilde{s}_a \geq \sigma$ is zero. Thus,

$$P(\sigma \in [\tilde{s}_a, \tilde{s}_b]) = P(\tilde{s}_a \leq \sigma \leq \tilde{s}_b) = \frac{1}{2} + \frac{s_b - \sigma}{2\mu}.$$

Case 3: $\sigma - \mu \leq s_a \leq \sigma + \mu \leq s_b$.

The probability that $\tilde{s}_b \leq \sigma$ is zero. Thus,

$$P(\sigma \in [\tilde{s}_a, \tilde{s}_b]) = P(\tilde{s}_a \leq \sigma \leq \tilde{s}_b) = \frac{1}{2} + \frac{s_a - \sigma}{2\mu}.$$

□

Consider a random edge with span $[x, x + \delta]$. To compute the probability that such an edge intersects the isosurface we integrate over all possible values of x . Let $\text{span}_{\tilde{G}}(e)$ be the span of edge e in \tilde{G} and let $E(G)$ be the set of edges of G .

Lemma 2. *Let \tilde{G} be a scalar grid created by adding random noise with uniform distribution in the range $[-\mu, \mu]$ to G . Let e be a random grid edge with span $[x, x + \delta]$ chosen from some $E' \subseteq E(G)$ such that x is uniformly distributed over $[\sigma - \Lambda, \sigma + \Lambda]$ for some $\Lambda \geq \mu + \delta$.*

1. *If $\delta \leq 2\mu$, then the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is*

$$\frac{8\mu^3 - \delta^3}{24\mu^2\Lambda} + \frac{\delta^2}{4\mu\Lambda}.$$

2. *If $\delta \geq 2\mu$, then the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is $\delta/(2\Lambda)$.*

Proof of Statement 1: Applying Lemma 1, Statement 1, the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ given that $\sigma - \mu \leq x \leq x + \delta \leq \sigma + \mu$ is:

$$\begin{aligned} \frac{1}{2\mu - \delta} \int_{x=\sigma-\mu}^{\sigma+\mu-\delta} \left(\frac{1}{2} - \frac{(x-\sigma)(x+\delta-\sigma)}{2\mu^2} \right) dx \\ = \frac{1}{2\mu - \delta} \left(\frac{2}{3}\mu - \frac{\delta^3}{12\mu^2} \right) = \frac{1}{2\mu - \delta} \left(\frac{8\mu^3 - \delta^3}{12\mu^2} \right). \end{aligned}$$

The probability that $\sigma - \mu \leq x \leq x + \delta \leq \sigma + \mu$ is $(2\mu - \delta)/(2\Lambda)$. The probability that $\sigma - \mu \leq x \leq x + \delta \leq \sigma + \mu$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\left(\frac{2\mu - \delta}{2\Lambda} \right) \left(\frac{1}{2\mu - \delta} \right) \left(\frac{8\mu^3 - \delta^3}{12\mu^2} \right) = \frac{8\mu^3 - \delta^3}{24\mu^2\Lambda}.$$

Applying Lemma 1, Statement 2, the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ given that $x \leq \sigma - \mu \leq x + \delta \leq \sigma + \mu$ is:

$$\frac{1}{\delta} \int_{x=\sigma-\mu-\delta}^{\sigma-\mu} \left(\frac{1}{2} + \frac{x+\delta-\sigma}{2\mu} \right) dx = \frac{\delta}{4\mu}.$$

The probability that $x \leq \sigma - \mu \leq x + \delta \leq \sigma + \mu$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\left(\frac{\delta}{2\Lambda}\right) \left(\frac{\delta}{4\mu}\right) = \frac{\delta^2}{8\mu\Lambda}.$$

Similarly, the probability that $\sigma - \mu \leq x \leq \sigma + \mu \leq x + \delta$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is also $\delta^2/(8\mu\Lambda)$.

Adding the three cases, $\sigma - \mu \leq x \leq x + \delta \leq \sigma + \mu$ and $x \leq \sigma - \mu \leq x + \delta \leq \sigma + \mu$ and $\sigma - \mu \leq x \leq \sigma + \mu \leq x + \delta$, the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\frac{8\mu^3 - \delta^3}{24\mu^2\Lambda} + \frac{\delta^2}{8\mu\Lambda} + \frac{\delta^2}{8\mu\Lambda} = \frac{8\mu^3 - \delta^3}{24\mu^2\Lambda} + \frac{\delta^2}{4\mu\Lambda}.$$

□

Proof of Statement 2: If σ is in $\text{span}_{\tilde{G}}(e)$, then x must be in the range $[\sigma - \mu - \delta, \sigma + \mu]$. We divide this range into three cases:

$x \in [\sigma - \mu - \delta, \sigma + \mu - \delta]$ and $x \in [\sigma + \mu - \delta, \sigma - \mu]$ and $x \in [\sigma - \mu, \sigma + \mu]$.

Applying Lemma 1, Statement 2, the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ given that $\sigma - \mu - \delta \leq x \leq \sigma + \mu - \delta$ is:

$$\frac{1}{2\mu} \int_{x=\sigma-\mu-\delta}^{\sigma+\mu-\delta} \left(\frac{1}{2} + \frac{x + \delta - \sigma}{2\mu}\right) dx = \frac{1}{2}.$$

The probability that $\sigma - \mu - \delta \leq x \leq \sigma + \mu - \delta$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is $\mu/(2\Lambda)$.

The probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ given that $\sigma + \mu - \delta \leq x \leq \sigma - \mu$ is 1 since $\mu \leq \delta/2$. The probability that $\sigma + \mu - \delta \leq x \leq \sigma - \mu$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is $(\delta - 2\mu)/(2\Lambda)$.

The probability that $\sigma - \mu \leq x \leq \sigma + \mu$ and $\sigma \in \text{span}_{\tilde{G}}(e)$ is symmetric to the first case and is also $\mu/(2\Lambda)$.

Summing the three cases, the total probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is $\delta/(2\Lambda)$. □

Now consider a random edge e . To compute the probability that e is intersected by the isosurface we integrate over the magnitude of the edge spans.

Lemma 3. Let \tilde{G} be a scalar grid created by adding random noise with uniform distribution in the range $[-\mu, \mu]$ to G . Let e be a random grid edge with span $[x, x + \delta]$ chosen from $E' \subseteq E(G)$ such that Ψ restricted to E' has bivariate uniform distribution on $[\sigma - \Lambda, \sigma + \Lambda] \times [0, \gamma]$ for some $\Lambda \geq \mu + \gamma$.

1. If $\gamma \leq 2\mu$, then the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\frac{32\mu^3 + 8\mu\gamma^2 - \gamma^3}{96\mu^2\Lambda}.$$

2. If $\gamma \geq 2\mu$, then the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\frac{3\gamma^2 + 2\mu^2}{12\gamma\Lambda}.$$

Proof of Statement 1: Applying Lemma 2 and integrating over δ , the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ is:

$$\frac{1}{\gamma} \int_{\delta=0}^{\gamma} \left(\frac{8\mu^3 - \delta^3}{24\mu^2\Lambda} + \frac{\delta^2}{4\mu\Lambda} \right) d\delta = \frac{32\mu^3 + 8\mu\gamma^2 - \gamma^3}{96\mu^2\Lambda}.$$

□

Proof of Statement 2: We integrate the probability that $\sigma \in \text{span}_{\tilde{G}}(e)$ over δ , splitting the integrand into two cases, one where $\delta \leq 2\mu$ and the other where $\delta \geq 2\mu$. Applying Lemma 2, the probability is:

$$\begin{aligned} \frac{1}{\gamma} \left(\int_{\delta=0}^{2\mu} \left(\frac{8\mu^3 - \delta^3}{24\mu^2\Lambda} + \frac{\delta^2}{4\mu\Lambda} \right) d\delta + \int_{\delta=2\mu}^{\gamma} \left(\frac{\delta}{2\Lambda} \right) d\delta \right) \\ = \frac{1}{2\gamma} \left(\frac{7\mu^2}{3\Lambda} + \frac{\gamma^2}{2\mu} - \frac{2\mu^2}{\Lambda} \right) = \frac{1}{2\gamma} \left(\frac{\gamma^2}{2\Lambda} + \frac{\mu^2}{3\Lambda} \right) = \frac{3\gamma^2 + 2\mu^2}{12\gamma\Lambda}. \end{aligned}$$

□

Using Lemma 3 we can compute the expected number of intersected edges in a noisy grid \tilde{G} and compute the expected fractal edge span dimension. Thus we prove Theorem 1 by employing Lemma 3.

Proof of Theorem 1, Statement 1:

Let M be the number of edges in $E_{\sigma}(\eta)$. By assumption, $M/8$ equals $|E'_{\sigma}(\eta)|/8$. Let μ equal $r\gamma$. The added noise is uniformly distributed in the range $[-\mu, \mu]$. By Lemma 3, the expected value of $I_{\epsilon}^{\sigma}(\tilde{G})$, the number of edges of \tilde{G} intersecting the isosurface is: $M \left(\frac{32\mu^3 + 8\mu\gamma^2 - \gamma^3}{96\mu^2\Lambda} \right)$.

Let \tilde{G}' be the subsampled regular grid of \tilde{G} with $N/8$ vertices. Since each subgrid edge covers two edges of grid \tilde{G} , the magnitudes of edge spans of \tilde{G}' vary uniformly from 0 to 2γ . Thus the expected number of edges of \tilde{G}' intersecting the isosurface is: $(M/8) \left(\frac{32\mu^3 + 8\mu(2\gamma)^2 - (2\gamma)^3}{96\mu^2\Lambda} \right)$.

Dividing the number for the full grid by the number for the subsampled grid and replacing μ by $r\gamma$ gives

$$\begin{aligned} 8 \left(\frac{32\mu^3 + 8\mu\gamma^2 - \gamma^3}{32\mu^3 + 8\mu(2\gamma)^2 - (2\gamma)^3} \right) &= 8 \left(\frac{32(r\gamma)^3 + 8(r\gamma)\gamma^3 - \gamma^3}{32(r\gamma)^3 + 8(r\gamma)(2\gamma)^2 - (2\gamma)^3} \right) \\ &= 8 \left(\frac{32r^3 + 8r - 1}{32r^3 + 32r - 8} \right) = \frac{32r^3 + 8r - 1}{32(r/2)^3 + 8(r/2) - 1}. \end{aligned}$$

Thus the expected fractal dimension is: $\log_2 \left(\frac{32r^3 + 8r - 1}{32(r/2)^3 + 8(r/2) - 1} \right)$.

□

Proof of Theorem 1, Statement 2:

As in the proof of Statement 2, the number of edges of \tilde{G} intersecting the isosurface is: $M \left(\frac{32\mu^3 + 8\mu\gamma^2 - \gamma^3}{96\mu^2\Lambda} \right)$.

Let \tilde{G}' be the subsampled regular grid of \tilde{G} with $N/8$ vertices. The magnitudes of edge spans of \tilde{G}' vary uniformly from 0 to 2γ . Since $2\gamma > 2\mu$, Statement 2 of Lemma 3 applies, and the expected number of edges of \tilde{G}' intersecting the isosurface is: $(M/8) \left(\frac{3(2\gamma)^2 + 2\mu^2}{12(2\gamma)\Lambda} \right)$.

Dividing the number for the full grid by the number for the subsampled grid, replacing μ by $r\gamma$, and taking the logarithm gives Equation 3.3. \square

Proof of Theorem 1, Statement 3:

By Statement 2 of Lemma 3, the expected number of edges of \tilde{G} intersecting the isosurface is: $M \left(\frac{3\gamma^2 + 2\mu^2}{12\gamma\Lambda} \right)$.

Let \tilde{G}' be the subsampled regular grid of \tilde{G} with $N/8$ vertices. The expected number of edges of \tilde{G}' intersecting the isosurface is: $(M/8) \left(\frac{3(2\gamma)^2 + 2\mu^2}{12(2\gamma)\Lambda} \right)$.

Dividing the number for the full grid by the number for the subsampled grid, replacing μ by $r\gamma$, and taking the logarithm gives Equation 3.4. \square

3.2.1 Experimental Results

To verify our formula we created two synthetic data sets that represented the Euclidean distance from the origin. The first data set represented the distance to multiple points while the second data set was simply the distance to a circle. We added uniform noise in the range $[-\mu, \mu]$ to the scalar grid to create a noisy grid. We used various values for $\mu \in [0 : 4]$. For $\mu \geq 1$ Equation 1 applies, for $0.5 \leq \mu < 1$ Equation 2 applies, and for $\mu < 0.5$ Equation 3 applies. We computed the fractal edge span dimension for both of these data sets at increasing values of μ and, as shown in Figure 3.5, found that they match our predicted values.

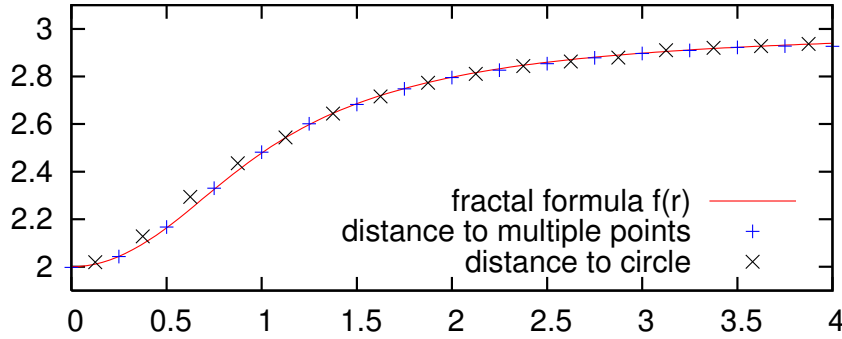


Figure 3.5: Plot of fractal edge span dimension versus noise added to a data set representing Euclidean distances to a set of 10 random points. Plot of fractal edge span dimension versus noise added to a data set representing distance to a circle. Plot of fractal formula.

The first of these data sets, the distance to multiple points, satisfies the assumptions that we made in our derivation. However the second data set is the scaled distance to a circle, the isosurfaces of which are tori. The edge spans of this data set are not uniformly distributed and thus do not satisfy the conditions of Theorem 1. Nevertheless, the formula does a good job of predicting the fractal dimension.

Nearly all benchmark data sets fail to satisfy the conditions of our formula. However for certain isovalues in certain data sets the necessary conditions are approximated well enough that the fractal formula can still provide a good prediction of the fractal dimension. We looked for

isosurfaces where the gradient magnitudes were constant in a neighborhood of the isosurface. If the gradient directions vary uniformly in all directions, then the magnitude of edge spans vary uniformly from 0 to γ . We found two such isosurfaces that approximated these conditions, Tooth around isovalue 650 (+/- 50) and Boston Teapot around isovalue 65 (+/- 15). For Tooth at isovalue 650 the edge spans vary uniformly from 0 to 120. For Boston Teapot at isovalue 65 the edge spans vary uniformly from 0 to 25. For each data set and range, the minimum edge scalar values also vary uniformly.

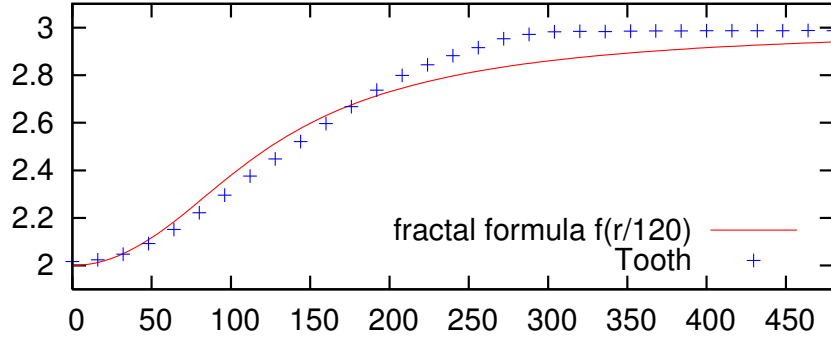


Figure 3.6: Plot of fractal dimension (isovalue 650) vs. noise added to the Tooth data set and plot of fractal formula adjusted for the gradient 120.

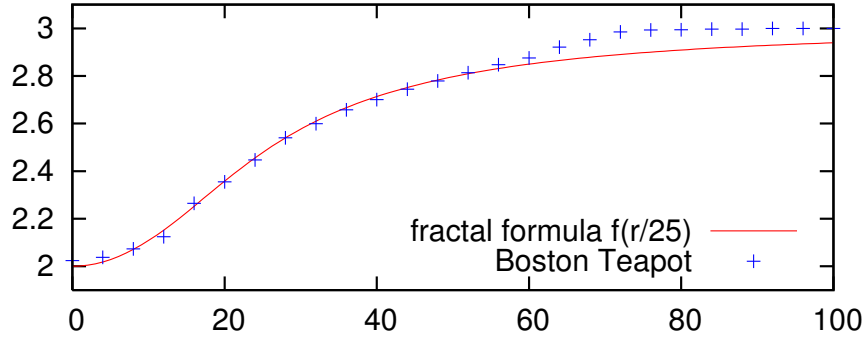


Figure 3.7: Plot of fractal dimension (isovalue 65) vs. noise added to the Boston Teapot data set and plot of fractal formula adjusted for the gradient 25.

Figures 3.6 and 3.7 show the fractal edge span dimension as increasing amounts of uniform noise is added to the scalar grids. As can be seen from the figures, the fractal formula approximates the correct fractal dimension, although not as well as for the synthetic data sets.

Chapter 4

Filtering

Isosurface construction is a popular technique for visualizing scalar data. However it has the disadvantage of being extremely susceptible to noise present in the data. To cope with this drawback, many different filtering techniques have been applied to scalar data to remove noise and produce smooth isosurfaces. Median and Gaussian filters are two commonly used filters, while the more recent bilateral filter is also popular. We propose a novel technique based on our scheme for noise identification.

4.1 Median Filtering

The median is a nonlinear digital filtering technique that is commonly used in signal processing to reduce noise. The algorithm proceeds entry by entry along the signal, replacing each value with the median of its neighbors. In 1D the definition of neighbors is obvious, but in 2D more complicated windows can be used, such as the Moore or von Neumann neighborhoods.

Applying this filter to the scalar grids used in isosurface construction is trivial. Processing each vertex v of the grid, we replace the scalar value s_v with the median scalar value of the neighbors of v . In our implementation we consider all 27 vertices in the immediate neighborhood.

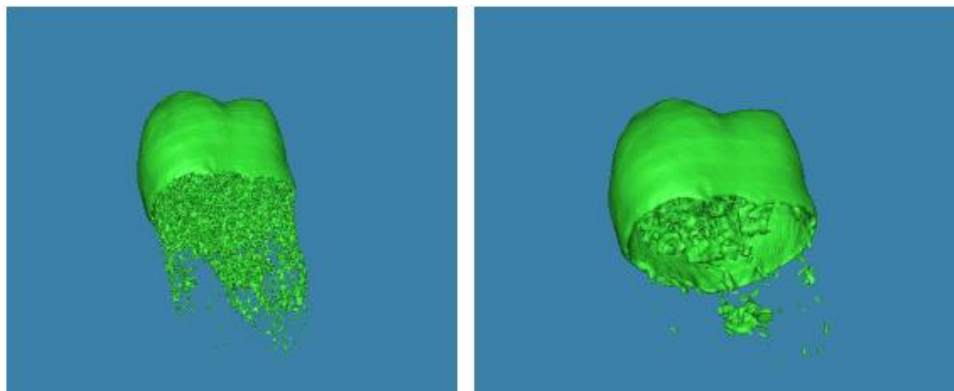


Figure 4.1: An isosurface for isovalue 850 extracted from the Tooth data set before(left) and after(right) a median filter was applied.

A median filter is very good at removing noise from a scalar grid. As demonstrated in Figure 4.1, the majority of small connected components have been removed from the isosurface extracted from the filtered grid.

4.2 Gaussian Filtering

The Gaussian filter proceeds in an identical manner to the median filter. The difference is in how the new scalar value s_v for the vertex v is chosen. Instead of setting s_v to the median of v 's neighbors, s_v is set to a weighted average of v 's neighbors. Different neighbors are assigned different weights, with the closest neighbors having the greatest weight. The weights are chosen from a Gaussian distribution.

Each neighbor is multiplied by its corresponding weight and summed. The total sum is then divided by the sum of the weights and the resulting value is assigned to s_v .

4.3 Fractal Filtering

Applying a filter to an entire data set smooths regions that may contain important features of a surface, as shown in Figure 4.2. Noise filters do not utilize any information about the structure that is being smoothed and simply smooth all the data they process. Important features, such as sharp corners, may be smoothed after a filter is applied to a scalar grid. If a filter knew which areas needed to be smoothed it could preserve important features and produce a result more quickly.

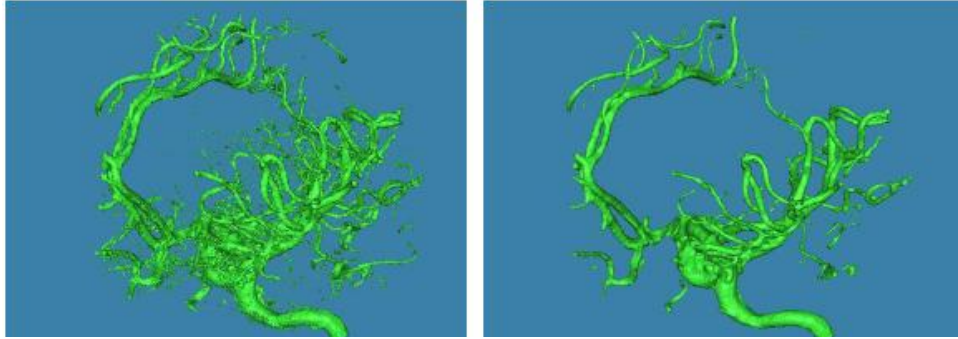


Figure 4.2: An isosurface for isovalue 20 extracted from the Aneurism data set before(left) and after(right) a median filter was applied. Since the entire grid was filtered, a great amount of detail was lost in the capillaries.

We propose a targeted filtering technique that utilizes the local fractal dimension around each grid vertex. First we compute the local fractal dimension and mark a vertex for filtering only if the fractal dimension is higher than some user specified parameter. We then apply a median or Gaussian filter to all marked vertices. Vertices that are in smooth regions are never touched by our filter, thus preserving as much of the original data as possible.

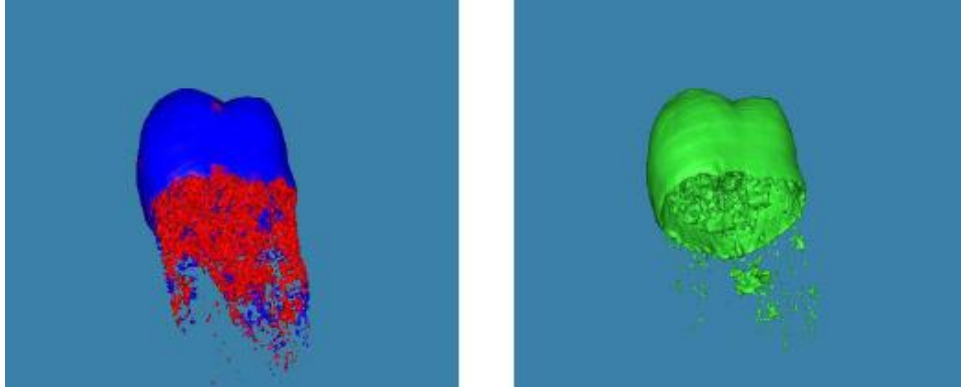


Figure 4.3: Using the local fractal dimension to identify the noisy regions in a scalar grid, scalar grid vertices are marked for filtering. A median filter is then applied only to the marked vertices. While only a portion of the grid was filtered there is no apparent decrease in quality in the isosurface.

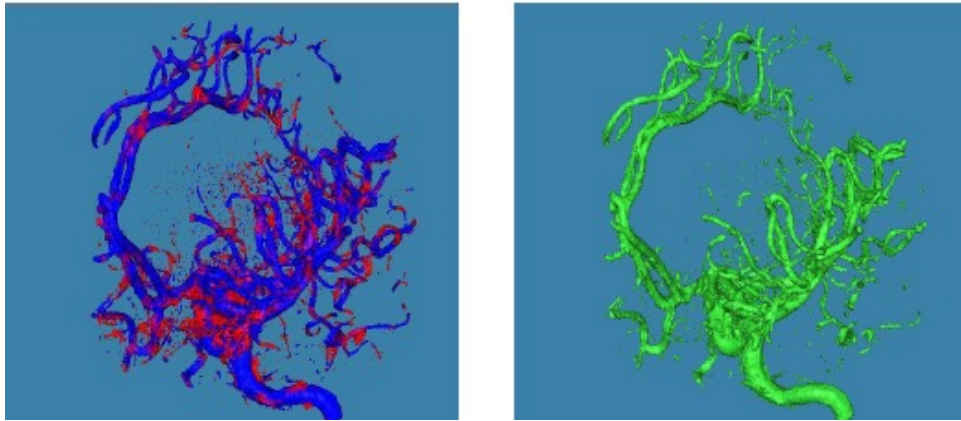


Figure 4.4: The local fractal dimension was used to identify the noise in the scalar grid and a median filter was applied only in the noisy regions. The resulting isosurface is much smoother than the original but preserves much more detail in the capillaries than if we filtered the entire grid.

Figures 4.3 and 4.4 demonstrate our selective filtering technique. A median filter is applied only to the noisy regions in the scalar grid. Comparing 4.3 to 4.1, there is no noticeable decrease in filter quality. However comparing 4.4 and 4.2, it is apparent that only filtering the noisy regions removes many of the small connected components from Aneurism and also preserves much more detail in the capillaries that was lost when a median filter was applied to the entire grid.

Computing the local fractal dimension across a scalar grid, we can efficiently determine which areas require filtering. Applying filtering only to these areas, we can remove the noise from the scalar grid, reduce the computational cost and preserve important features that may have been removed by other filters.

Chapter 5

Smooth Isosurface Construction

Isosurfaces are very susceptible to noise present in scalar data sets. The Marching Cubes algorithm frequently produces isosurfaces with numerous small components that are representative of noise in the data. Noise added to the scalar grid vertices creates new intersected edges and thus additional components for a level set $f^{-1}(\sigma)$. We would like to develop an algorithm that identifies these high fractal edges and alters their scalar values so that they are no longer intersected by the isosurface.

The algorithm starts by generating a local fractal grid for isovalue σ that will be used to identify grid edges that rest in high fractal regions. Algorithm 1 traverses the edges of the scalar grid. If a grid edge is intersected by an isosurface with isovalue σ and one of the two vertices of the edge has high local fractal dimension then the edge is pushed onto a queue. After the entire grid is traversed, the queue E contains all the high fractal edges. The run time of this algorithm is $\Theta(m)$, where m is the number of edges.

Algorithm 1 HighFractalEdges(Grid g , int σ , float h)

```
E  $\leftarrow$  empty queue of edges
for each grid edge  $e = (v, u)$  do
    if  $\sigma \in \text{span}(e)$  and ( $\text{dim}(v) > h$  or  $\text{dim}(u) > h$ ) then
        E.push( $e$ )
    end if
end for
return E
```

Once the high fractal edges are computed, Algorithm 2 begins to process the edges one at a time. Each edge e has two vertices (v, u) each with its own scalar value. Suppose that the scalar value $s_v > s_u$. The scalar value at v is set to the scalar value at u . Since both scalar values are now less than σ the edge e is no longer intersected by the isosurface. Changing the scalar value s_v potentially creates new intersected edges incident to v . These new intersected edges are examined for high fractal dimension. If a new edge lies in a high fractal dimension region it is pushed onto the queue for processing. The algorithm expands the — vertices (i.e. those with scalar value below σ), effectively moving the isosurface away from high fractal regions. The algorithm traverses the grid in a manner analogous to breadth-first search and only does a

constant amount of work per edge, thus the run time is $\Theta(n + m)$, where n is the number of vertices and m is the number of edges. Additionally termination is guaranteed, as each edge can be processed at most once. After processing, an edge is no longer intersected by the isosurface and no future operation performed by the algorithm will allow it to be intersected again. Thus it can never again be pushed into the queue. Upon termination each intersected edge rests in a low fractal region.

Algorithm 2 SmoothSurface(Grid g , int σ , float h)

```

Q  $\leftarrow$  HighFractalEdges( $g$ ,  $\sigma$ ,  $h$ )
while !Q.empty() do
    ( $v$ ,  $u$ )  $\leftarrow$  Q.pop()
     $g[v] \leftarrow g[u]$  //Edges are stored such that scalar value  $g[v] > g[u]$ 
    for each new intersected edge  $e$  around  $v$  do
        if  $e$  has high fractal dimension then
            Q.push( $e$ )
        end if
    end for
end while

```

There is a problem with setting the scalar value at v to that at u . The resulting isosurface tends to be very blocky around the modified regions because the scalar values change abruptly. Instead we not only modify s_v , but also the scalar values around v in a Gaussian manner. Compute $\Delta = s_v - s_u$ and decrease s_v and all of its immediate neighbors by some fraction of Δ . The scalar value s_v is reduced by exactly Δ resulting in the value s_u . The neighbors of v are decreased by fractions of Δ based on a Gaussian distribution. Thus the further away vertices are decreased by a smaller fraction of Δ than those closer to v . Decreasing the neighbors in this smooth manner reduces produces smoother isosurfaces.

5.1 Experimental Results

Figure 5.1 shows Algorithm 2 applied to the Tooth data set for isovalue 850. Three different values for the h parameter have been used to demonstrate the sensitivity of the parameter. The isosurface on the left was generated from a scalar grid where 2.1 was used for the h fractal value, meaning that most of the surface lied in a high fractal region and needed to be moved. As can be seen, setting the parameter too low creates holes in the surface. The h value for the center image was 2.2. In this case the noise has been removed but so have a some of the lower portion of the tooth cap. The h value for the image on the right was 2.5. Most of the noise has been removed and only the smooth portion of the surface remains.

Examining the lower portion of the tooth cap, we notice that the surface appears blocky. These surfaces were generated by the version of Algorithm 2 that did not modify the neighbors of v . Figure 5.2 shows a different data set, using the modified algorithm.

The VisMale data set at isovalue 70 has regions where that represent the skull and regions that represent the skin. Figure 5.2 shows the unmodified isosurface on the left. Applying Algorithm 2

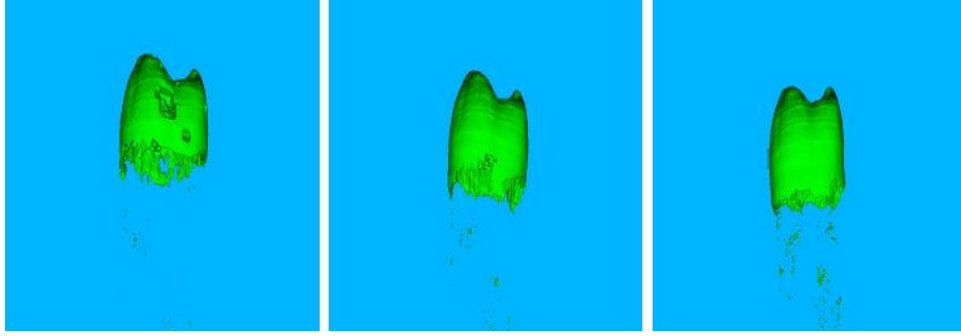


Figure 5.1: Algorithm 2 applied to the Tooth data set at isovalue 850. Various values for the parameter h determine how much of the surface is removed. From left to right these values are 2.1, 2.2, and 2.5.

with h parameter 2.2, all the skin along the back of the head is removed leaving only the skull, as shown in the center image. Unfortunately the algorithm has started to create holes in the surface. The image on the right uses a h parameter of 2.5, but fails to remove the skin from the back of the neck.

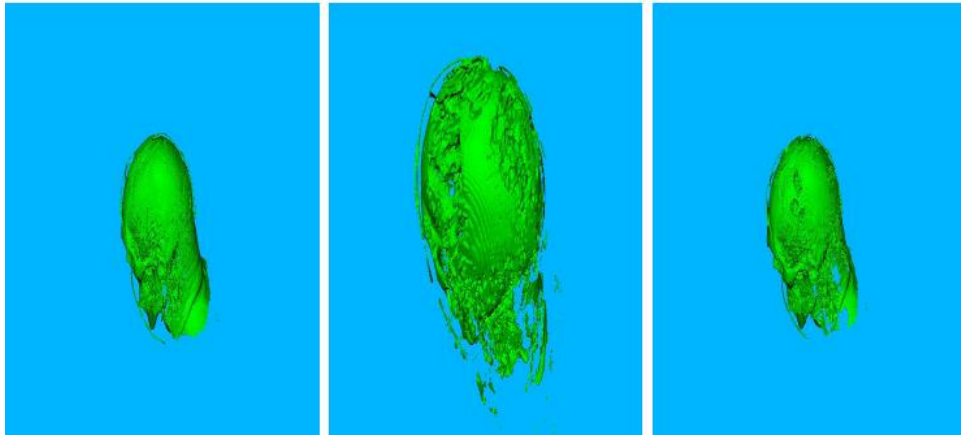


Figure 5.2: Algorithm 2 applied to the VisMale data set at isovalue 70. The image on the left is the isosurface generated by the Marching Cubes algorithm for VisMale at isovalue 70. The center image applies Algorithm 2 with h parameter 2.2. The right image uses value 2.5.

Great care needs to be applied in choosing the h parameter, as it defines how much of the surface is considered to lie in a noisy region. If the value is too low, a large portion of the surface will be removed. While this approach shows promise, more work needs to be done to avoid holes in the surface and to more effectively remove the noise from the surface.

Chapter 6

Conclusion

The fractal dimension is a powerful metric for measuring the complexity of isosurfaces. The fractal dimension can be quickly plotted as a function of scalar value and used for isovalue selection. We showed a strong positive correlation between the fractal dimension and the topological noise in the scalar grid by measuring the correlation for 60 benchmark data sets. To better understand the relationship between noise and fractal dimension we derived a formula that predicts the fractal dimension as a function of uniform noise in a data set.

Limiting our definition to a local region we demonstrated that the local isosurface fractal dimension can be used to identify noise in a scalar grid. This noise identification technique has a wide variety of applications, including selective filtering and smooth isosurface construction. We demonstrated how we can apply local fractal dimension to improve median and Gaussian filtering. Lastly we presented an algorithm to aid in smooth surface construction by moving an isosurface to a less noisy region of the grid.

Our work leaves many open problems. A short summary of these problems is presented to aid future work.

6.1 Intrinsic Fractal Dimension

The experiments show that at least part of the fractal dimension of an isosurface is caused by noise in the grid. However it remains an open questions if isosurfaces have any intrinsic fractal dimension.

6.2 Fractal Formula

The assumptions and noise model used to derive the formula presented in Section 3.2 is very limiting. Uniform noise is not a realistic noise model for real situations. What is the expected fractal dimension if we use a Gaussian noise model? Second we assume that the edge span magnitude are uniformly distributed in the neighborhood of the isovalue. This assumption does not hold for many isosurfaces. What is a more realistic model of the distribution of edge span magnitudes? Lastly the fractal edge span dimension and fractal box span dimension differ slightly, as demonstrated by Figure 3.4. What is the formula for the expected fractal box span dimension?

6.3 Filtering

Bilateral filtering[8] is an extension of Gaussian filtering that preserves the edges of an image. By examining the gradient a bilateral filter assures that only nearby pixels are used to compute a weighted average. Could we improve this filtering technique by using the fractal dimension instead of the gradient?

6.4 4D Fractal Dimension

The main disadvantage of our fractal filtering technique and smooth isosurface construction algorithm is that it is dependent on a particular isovalue used to compute the local fractal dimension. For each new isosurface a new local fractal grid must be computed. Instead, one can compute the 4D fractal dimension and similarly the 4D local fractal dimension. These metrics are independent of isovalue and produce just one single value for the entire grid instead of a value for each isovalue. Could we generate a single 4D local fractal grid and use it for filtering and smooth surface construction?

Bibliography

- [1] C L Bajaj, V Pascucci, and D R Schikore. The contour spectrum. In *Proceedings of the 8th conference on Visualization*, volume 0, pages 167–173. IEEE, 1997. ISBN 0818682620. doi: 10.1109/VISUAL.1997.663875. 1
- [2] H Carr, J Snoeyink, and U Axen. Computing contour trees in all dimensions. *Computational Geometry*, 24(2):75–94, 2003. ISSN 09257721. doi: 10.1016/S0925-7721(02)00093-7. 3.1
- [3] Hamish Carr, Brian Duffy, and Barry Denby. On histograms and isosurface statistics. *IEEE Transactions on Visualization and Computer Graphics*, 12(5):1259–1265, 2006. ISSN 10772626. doi: 10.1109/TVCG.2006.168. 1, 2.3
- [4] F Crow. Summed area tables for texture mapping. *Computer Graphics SIGGRAPH proceedings*, 11(3):200–220, 1984. 2.4
- [5] Kenneth Falconer. *Fractal Geometry: Mathematical Foundations and Applications*. Wiley, 2nd edition, 2003. ISBN 0470848626. 2.2
- [6] William E Lorensen and Harvey E Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics Forum*, 21(4):163–169, 1987. ISSN 00978930. doi: 10.1145/37402.37422. 1, 2.3
- [7] B B Mandelbrot. How long is the coastline of Britain? Statistical self-similarity and fractal dimension. *Science*, 155(636):636–638, 1967. 2
- [8] Sylvain Paris, Pierre Kornprobst, Jack Tumblin Tumblin, and Frédo Durand. Bilateral Filtering: Theory and Applications. *Foundations and Trends in Computer Graphics and Vision*, 4(1):1–75, 2008. ISSN 15722740. doi: 10.1561/06000000020. 6.3
- [9] V Pekar, R Wiemker, and D Hempel. Fast detection of meaningful isosurfaces for volume data visualization. *Proceedings of Visualization 2001*, D:223–230, 2001. doi: 10.1109/VISUAL.2001.964515. 2.3
- [10] Carlos E Scheidegger, John M Schreiner, Brian Duffy, Hamish Carr, and Cláudio T Silva. Revisiting Histograms and Isosurface Statistics. *IEEE Transactions on Visualization and Computer Graphics*, 14(6):1659–1666, 2008. ISSN 10772626. doi: 10.1109/TVCG.2008.160. 1, 2.3, 3.1